データマイニングと情報可視化

Week 7

稲垣 紫緒
いながき　しお

理学研究院　物理学部門 / 共創学部
inagaki@phys.kyushu-u.ac.jp
ウェスト1号館 W1-A823号室

授業計画

Week 1: Introduction
Week 2: Basics
データの入手
データ加工処理
可視化による見通し
分析結果の可視化
決定木
相関ルール
クラスタ分析
回帰分析
ニューラル
ネットワーク
Week 2 & 3
Week 4 & 5
Week 6 & 7
モデル選択
クロスバリデーション
Week 8: 応用

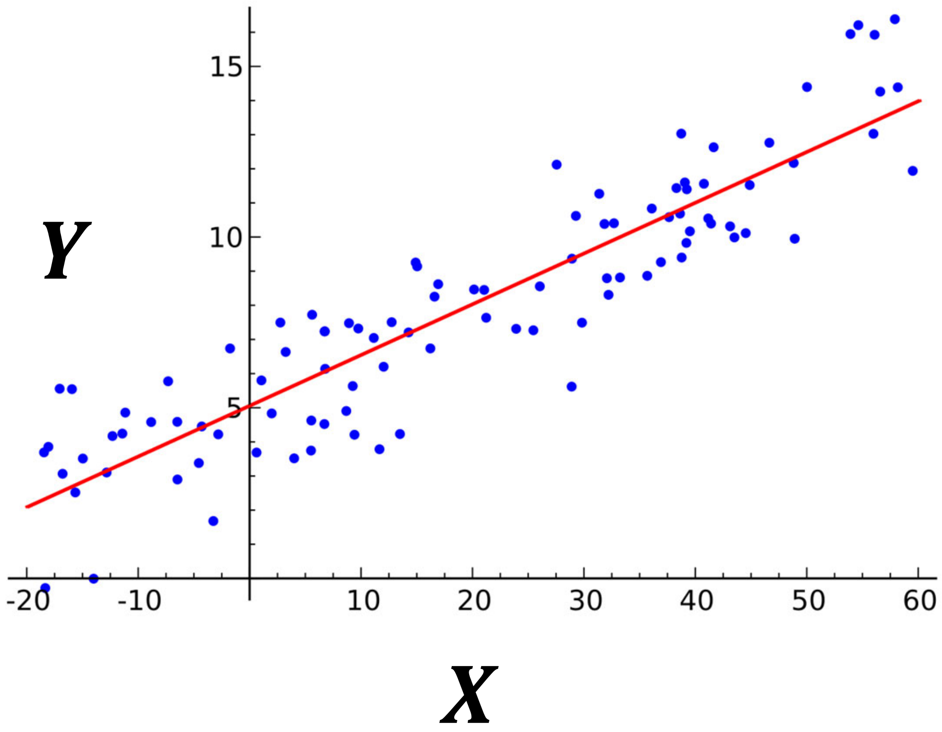
回帰分析による予測

Xの値から、 Yの値を予測したい。

→訓練データから、
　$\boldsymbol{\alpha}$と$\boldsymbol{\beta}$を推定する

$$Y = \alpha + \beta X$$

独立変数（説明変数） $\boldsymbol{X}$
従属変数（目的変数） $\boldsymbol{Y}$
の間にモデルを当てはめる。

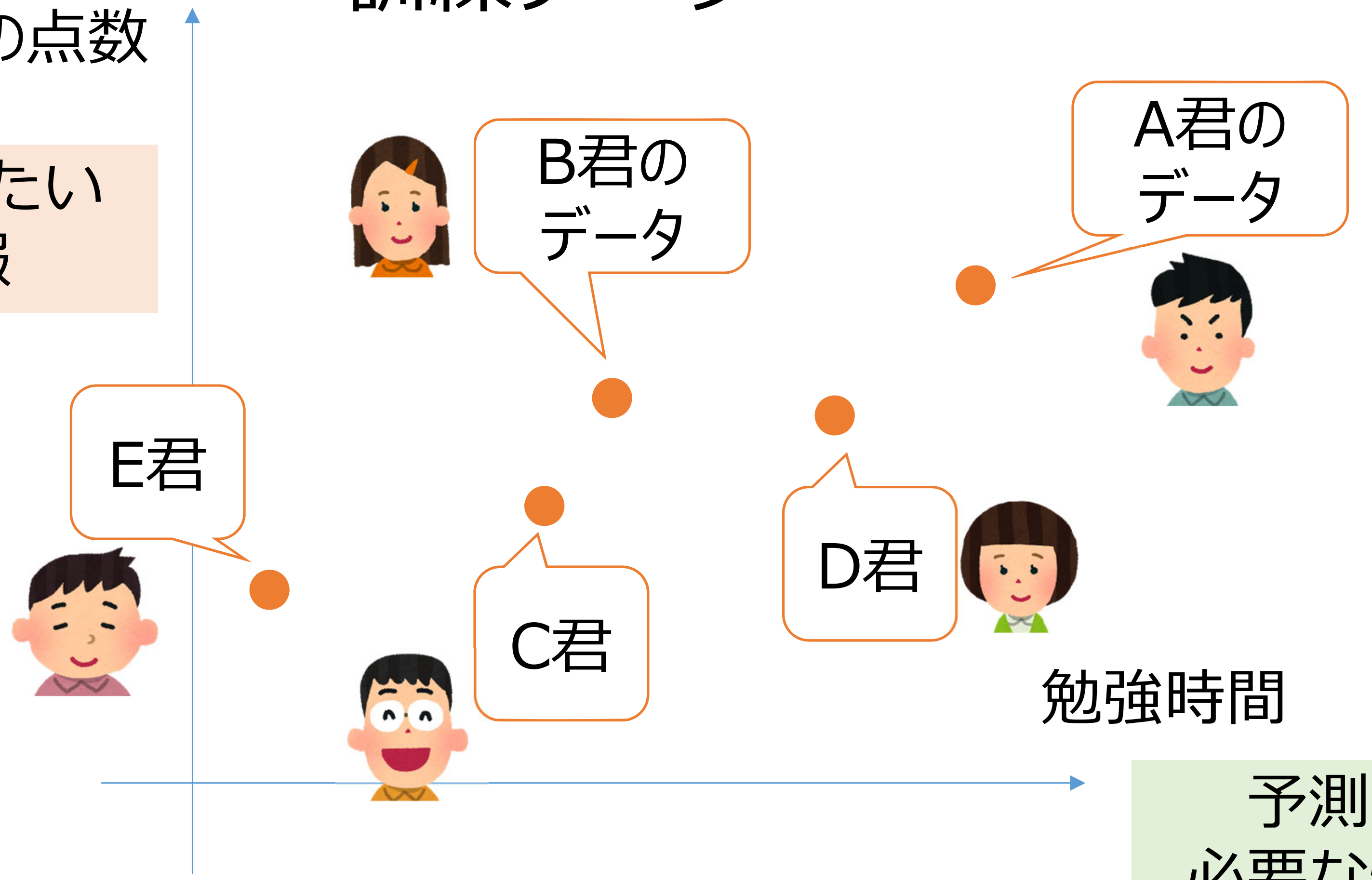


https://ja.wikipedia.org/wiki/%E5%9B%9E%E5%B8%B0%E5%88%86%E6%9E%90

回帰による予測(1/3) データ収集

復習

訓練データ



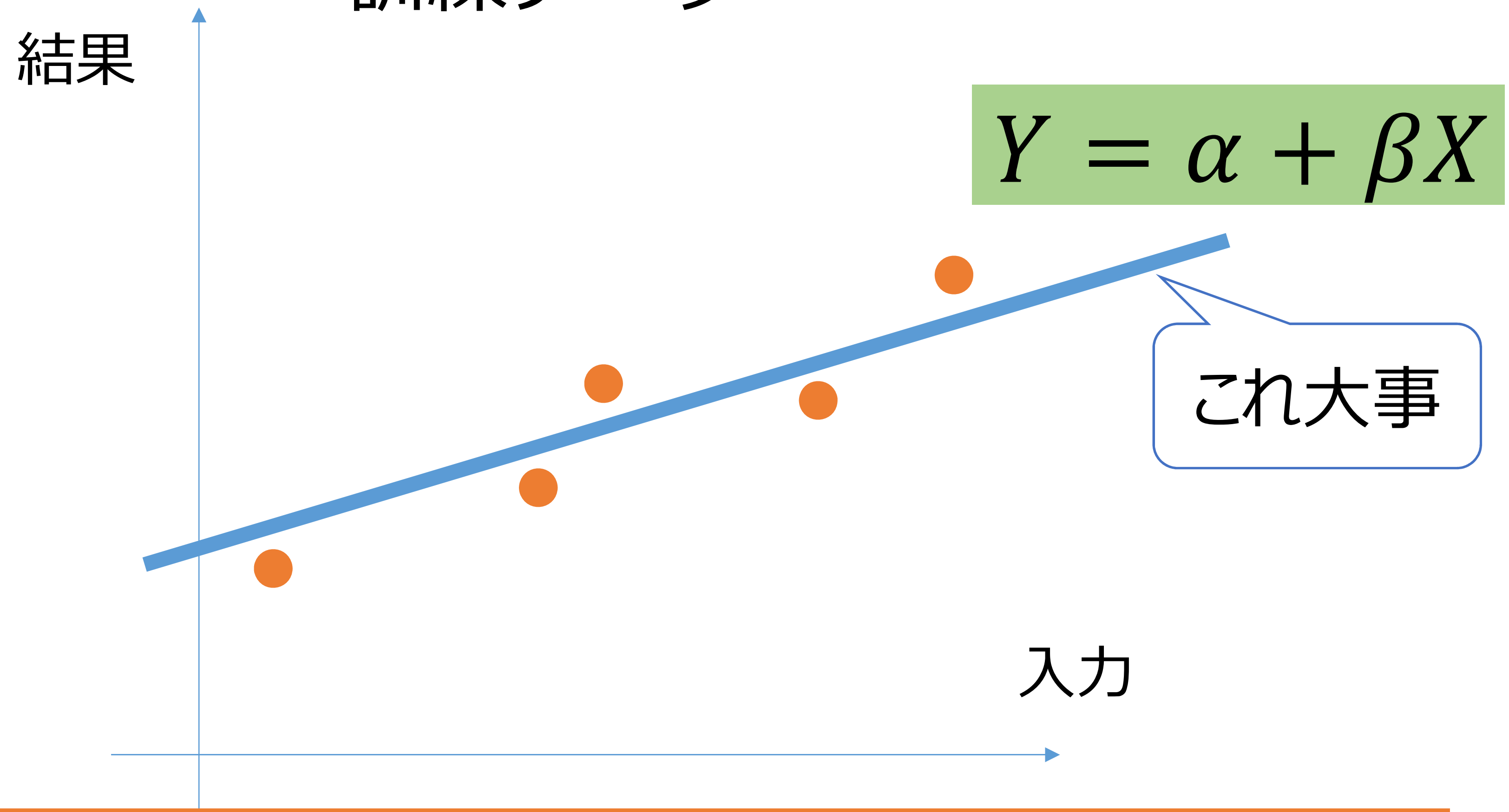
予測したい
情報

予測に
必要な情報

回帰による予測(2/3) モデル

復習

訓練データ



モデル＝「条件 or 入力」と「結果」間に成り立つと予想される関係．
上記は「線形モデル」

回帰による予測(3/3) 予測

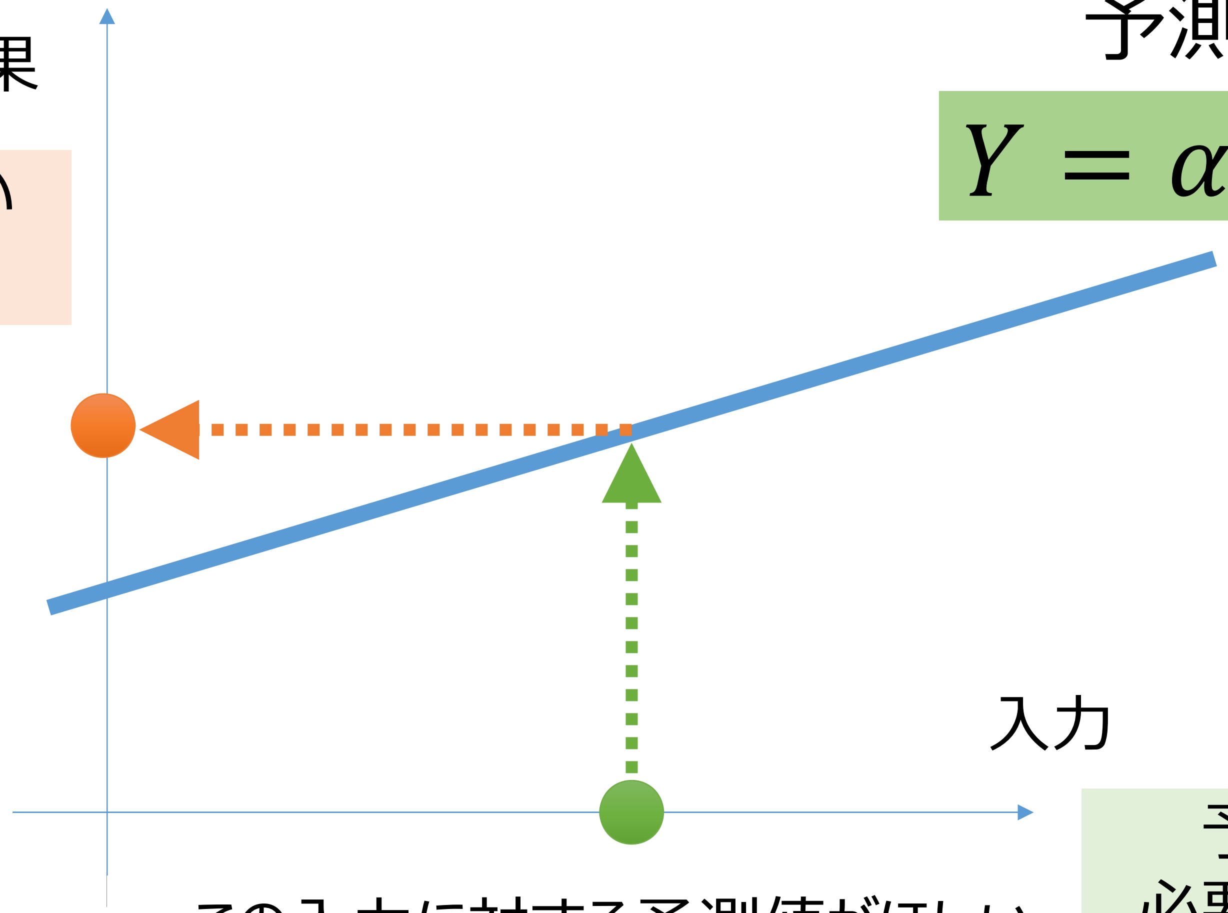
復習

予測器

$$Y = \alpha + \beta X$$

予測したい
情報

予測値
(65点)



予測に
必要な情報

この入力に対する予測値がほしい
(ex. 勉強3時間だと何点取れそう？)

回帰分析：線形回帰

復習

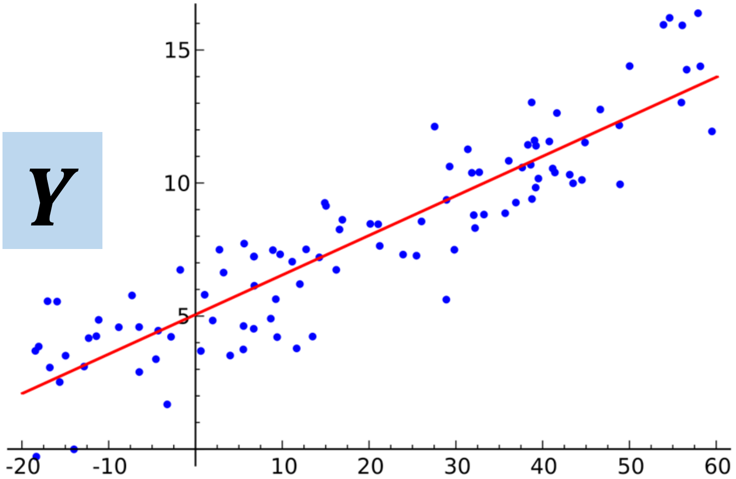
単回帰

説明変数が１つの場合

$$Y = \beta_o + \beta_1 X_1 + \epsilon$$

目的変数　　　説明変数

目的変数 Y



X 説明変数

重回帰

説明変数が複数の場合

$$Y = \beta_o + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots . + \beta_p X_p + \epsilon$$

$$Y = \beta_o + \beta_1 X_1 + \beta_1 X_2^2 + \epsilon$$

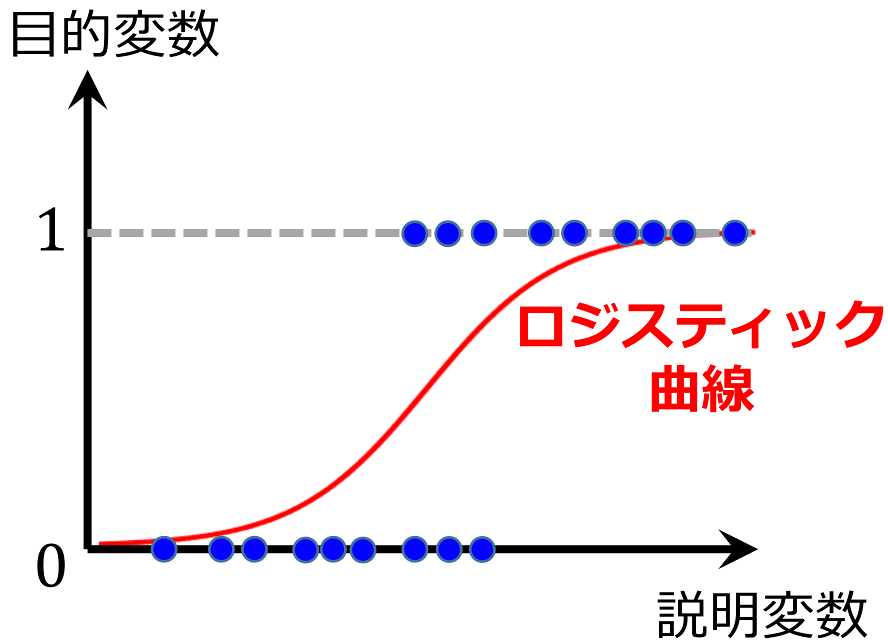
β_pについて線形(一次)だったら線形回帰

目的変数　　　説明変数

ロジスティック回帰分析

- ■ 迷惑メールかどうかの判定
- ■ 種子の発芽率
- ■ 病気の発病率・再発率
- ■ カブトムシの生存率
- ■ 企業の倒産確率
- ■ 商品の購買確率
- ■ 携帯の解約率
- ■ (内定の辞退率??)

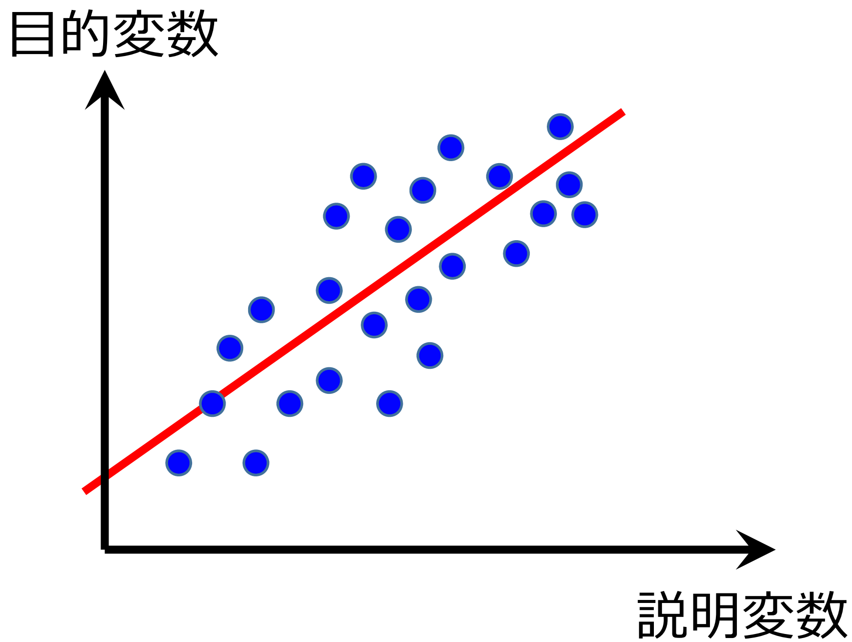
ロジスティック回帰分析
→**発生確率**の予測



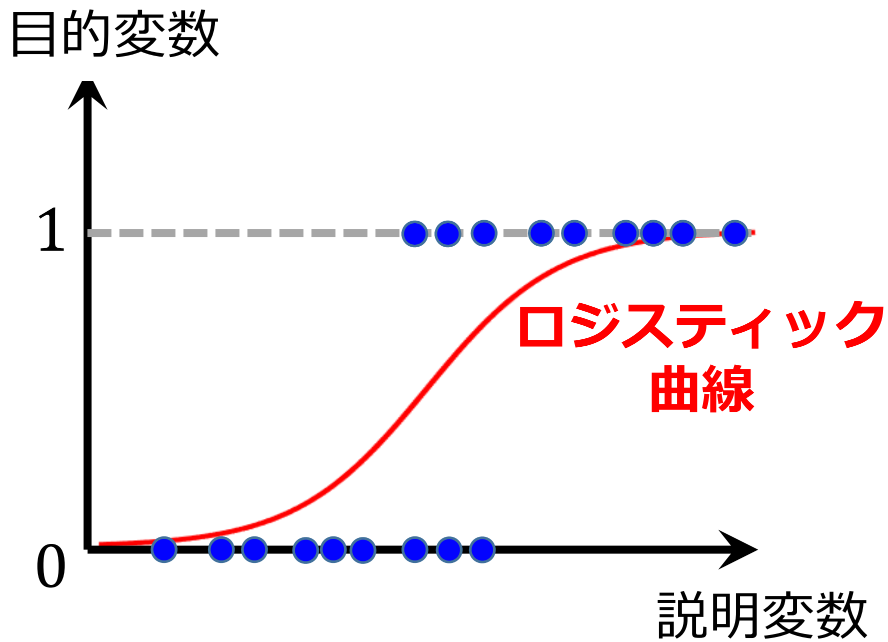
ある事象が **起きた→ 1**
起きなかった→0

ロジスティック回帰分析

線形回帰分析
→**量的変数**の予測



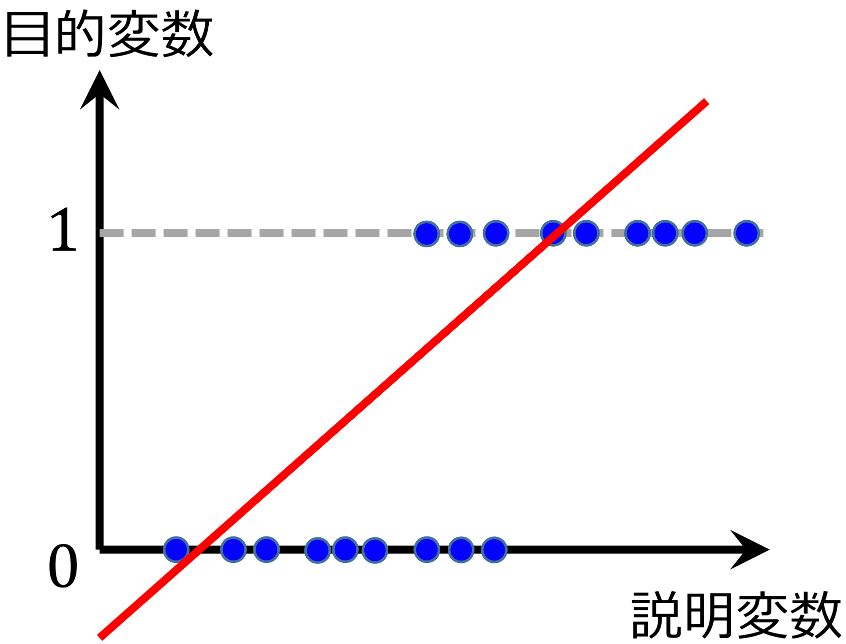
ロジスティック回帰分析
→**発生確率**の予測



ある事象が **起きた→ 1**
起きなかった→0

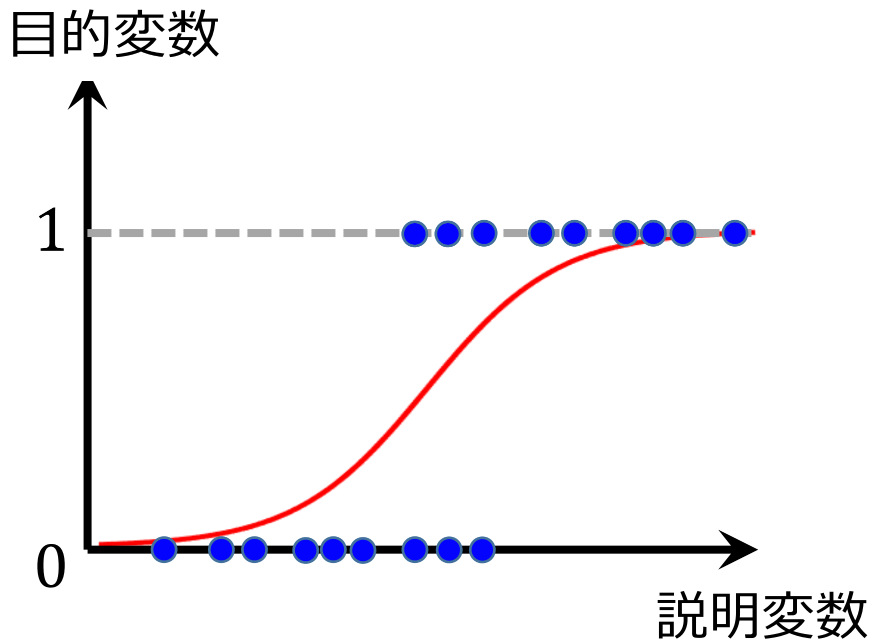
ロジスティック回帰分析

線形回帰で
連続値を予測しても意味がない。



「確率」なので、
3.5　とか　-1　とか
とりえない

ロジスティック回帰分析
→**発生確率**の予測



ある事象が **起きた→ 1**
起きなかった→0

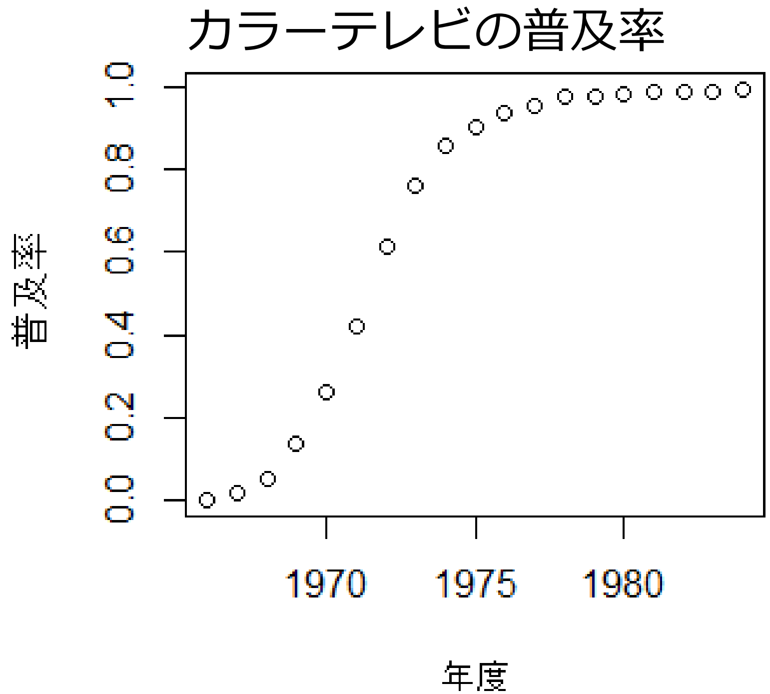
ロジスティック回帰分析

カラーテレビの普及率

経験的に、普及率や成長率のデータは
このような非線形曲線が多く、
ロジスティック関数にあてはまる。

$$y = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x))}$$

- β_0, β_1を求める。
- どれくらい実データと相関があるか、検証する



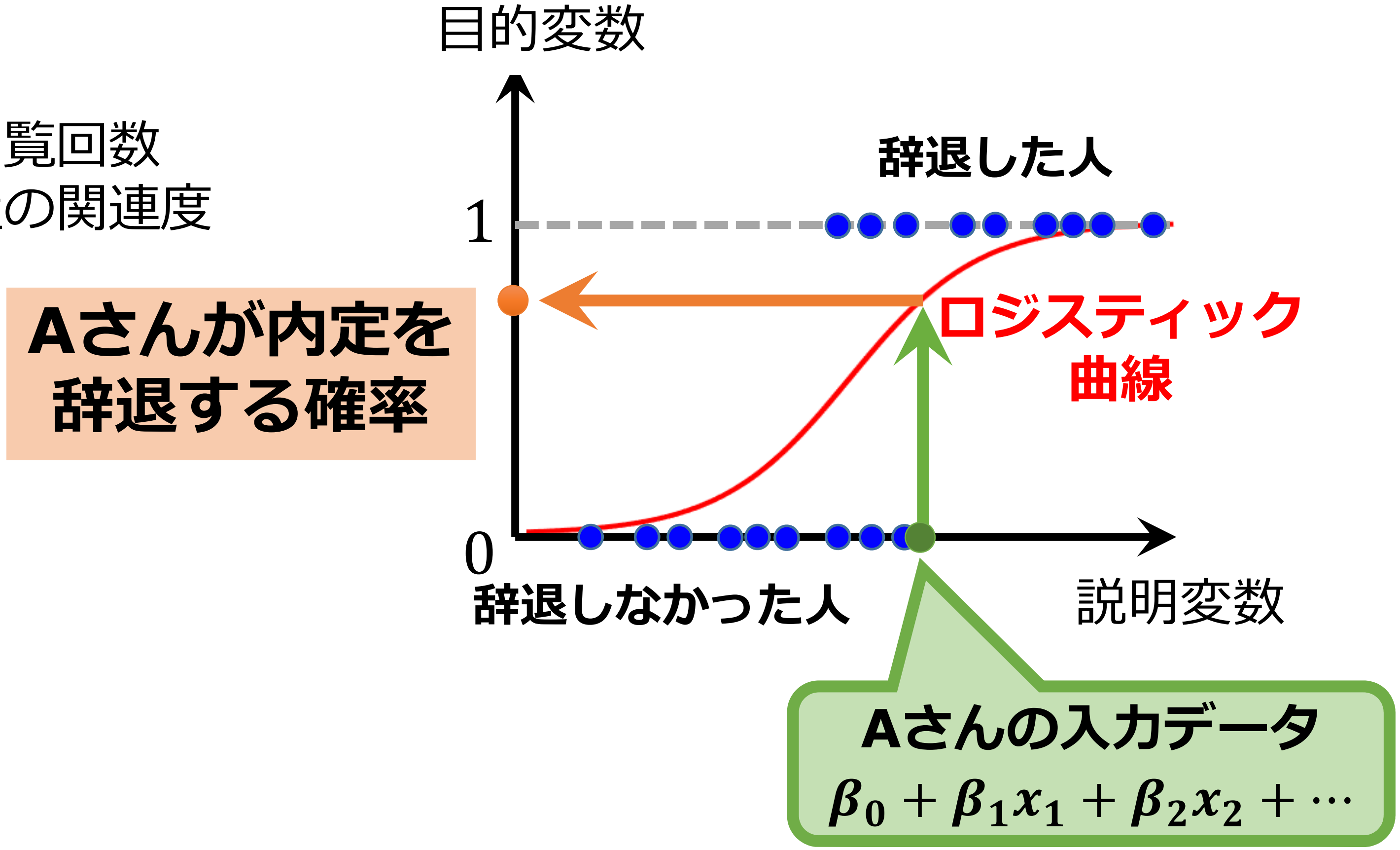
https://financial-it-engineer.hatenablog.com/entry/20140502/1398991089

ロジスティック回帰分析

ある事象が　**起きた→ 1**
起きなかった→0

説明変数
- ■ 企業ページの閲覧回数
- ■ その企業と自社の関連度

ロジスティック回帰分析
→**発生確率**の予測



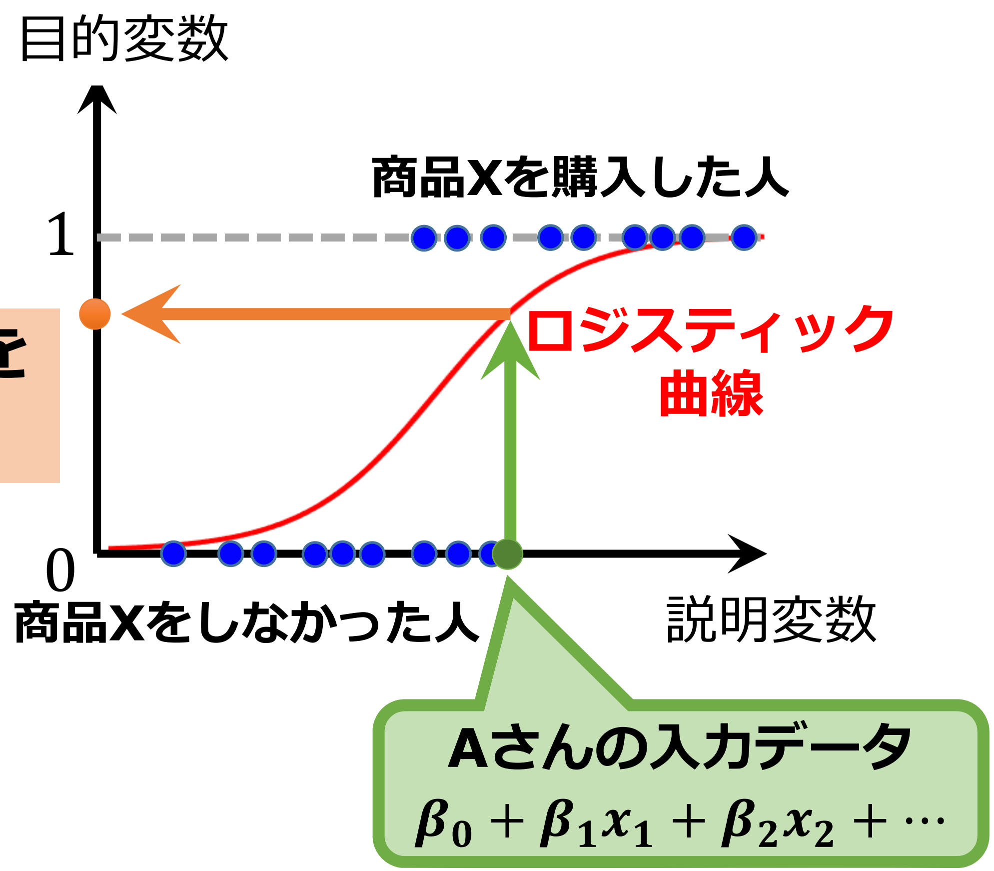
ロジスティック回帰分析

ある事象が　**起きた→ 1**
　　　　　　起きなかった→0

ロジスティック回帰分析
→**発生確率**の予測

説明変数
- 購入履歴
- その商品のサイトの閲覧回数
- 年収
- 職業
- 家族構成

Aさんが商品Xを購入する確率



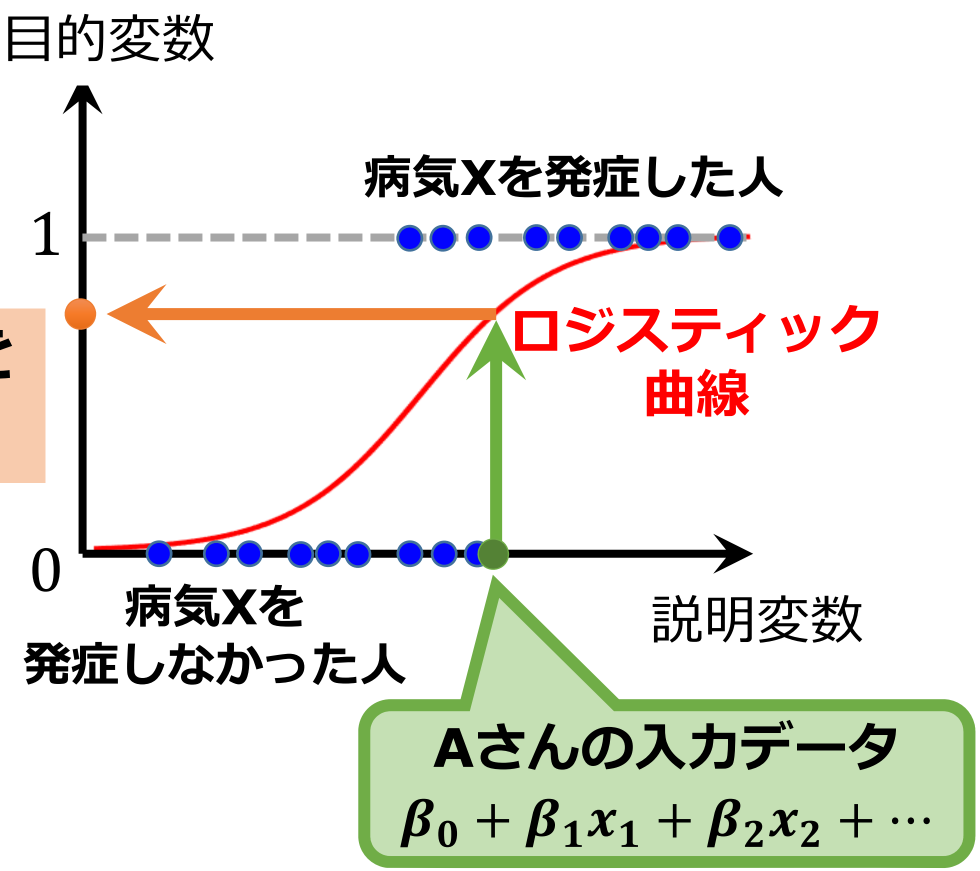
ロジスティック回帰分析

ある事象が **起きた→ 1**
起きなかった→0

ロジスティック回帰分析
→**発生確率**の予測

説明変数
- アルコール摂取量
- 喫煙歴
- 体脂肪率
- BMI
- 年齢

Aさんが病気Xを発症する確率



ロジスティック回帰分析

$$y = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i))}$$

- β_0, β_1を求める。
- どれくらい実データと相関があるか、検証する

偏回帰係数β_iの求め方

- ■ 最小二乗法
- ■ 最尤法

https://qiita.com/NaoyaOura/items/6ad5142b0306476d9293

ロジスティック回帰分析

$$y = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i))}$$

$$\frac{1 - y}{y} = \exp(-(\beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i))$$

ロジスティック回帰分析

$$y = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i))}$$

$$\frac{1 - y}{y} = \exp(-(\beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i))$$

$$\frac{y}{1 - y} = \exp(\beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i)$$

ロジスティック回帰分析

$$y = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i))}$$

$$\frac{1-y}{y} = \exp(-(\beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i))$$

$$\frac{y}{1-y} = \exp(\beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i)$$

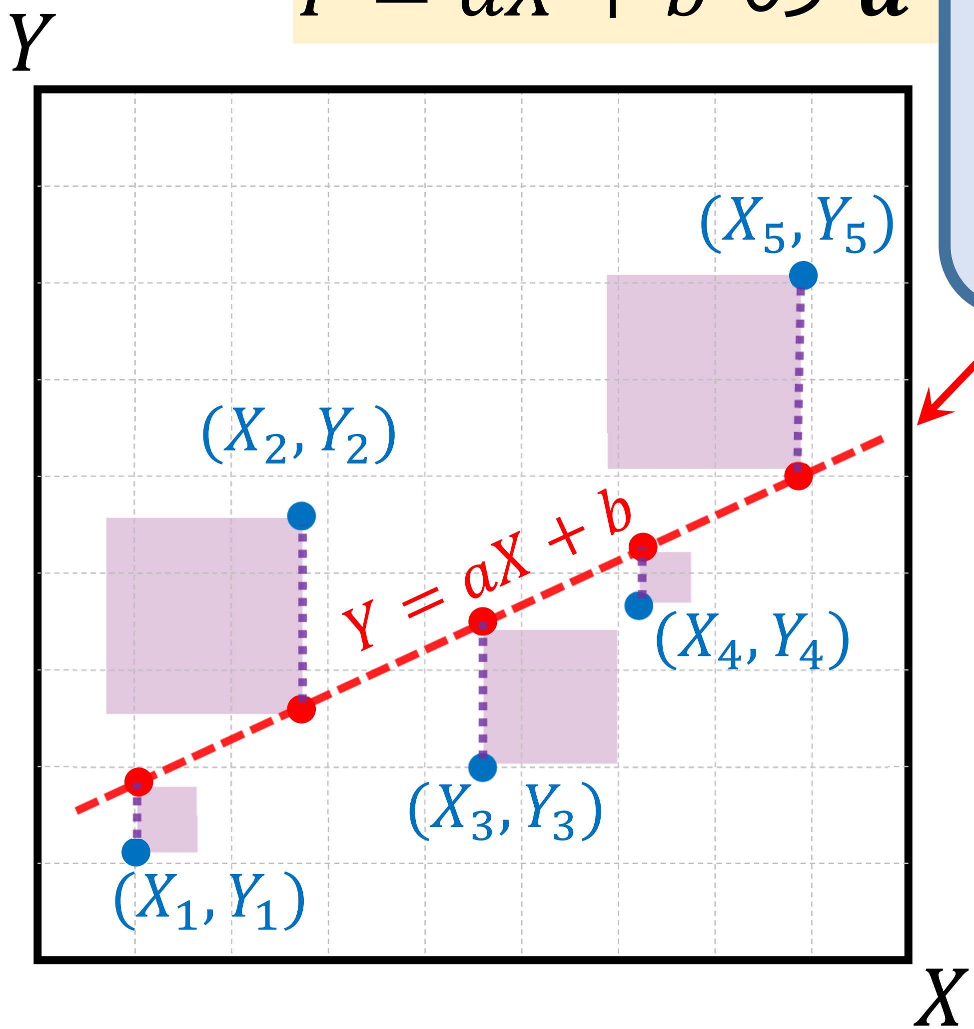
$$\ln\left(\frac{y}{1-y}\right) = \boxed{(\beta_0 + \beta_1 x_1 + \cdots + \beta_i x_i) = l}$$

重回帰分析と同じ!! $Y = \beta_o + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots . + \beta_p X_p + \epsilon$

最小二乗法

復習

$Y = aX + b$ の $\boldsymbol{a}$



E_{total}をXとYでそれぞれ偏微分
ともに0になる条件を
連立方程式で解く。

誤差の二乗の合計とが最小に
なるようなaとbを求める!!

誤差の2乗の合計

$$E_{\mathrm{total}} = \sum_{i=1}^{N} (Y_i - aX_i - b)^2$$

最小二乗法

導出の詳細は例えばこちら
https://manabitimes.jp/math/942

復習

まず直線の傾きを求める

$$a = \frac{s_{xy}}{\sigma^2} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$$

x と y の共分散 $\quad s_{xy} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})$

x の分散 $\quad \sigma^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})(x_i - \bar{x})$

オッズ比

$$\ln\left(\boxed{\frac{y}{1-y}}\right) = \boxed{(\beta_0+\beta_1 x_1 + \cdots + \beta_i x_i) = l}$$

l ：ロジット

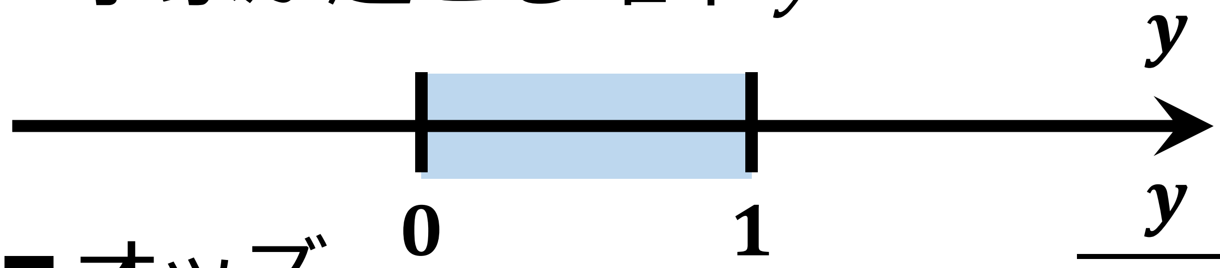
y ：事象の起こる確率

$1-y$ ：事象の起こらない確率

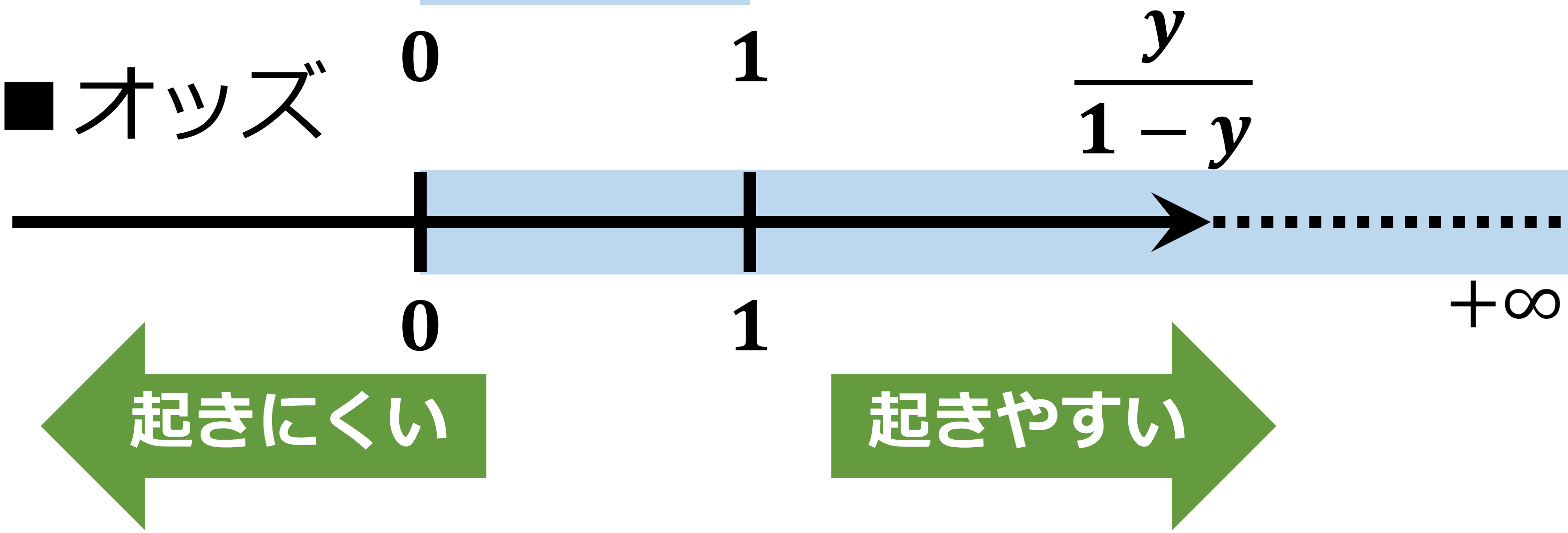
$\frac{y}{1-y} = \exp l$ ：オッズ

事象の起こりやすさの指標

■事象が起こる確率 y



■オッズ



$$y \ll 1 \Rightarrow \frac{y}{1-y} \sim y$$

ロジスティック回帰のオッズ

勝率	確率	オッズ	戻り金	
20%	0.2	0.25	1/0.2=5	5倍!!
50%	0.5	1	1/0.5=2	2倍
80%	0.8	4	1/8=1.25	1.25倍

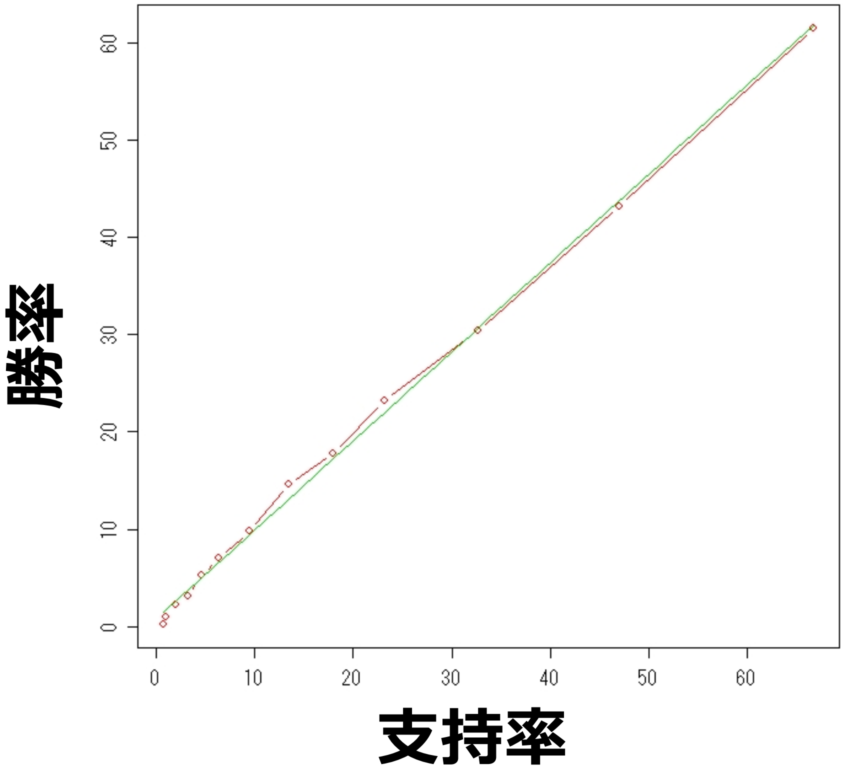
例) オッズ$\frac{y}{1-y} = \frac{0.2}{1-0.2} = \frac{0.2}{0.8} = \mathbf{0.25}$

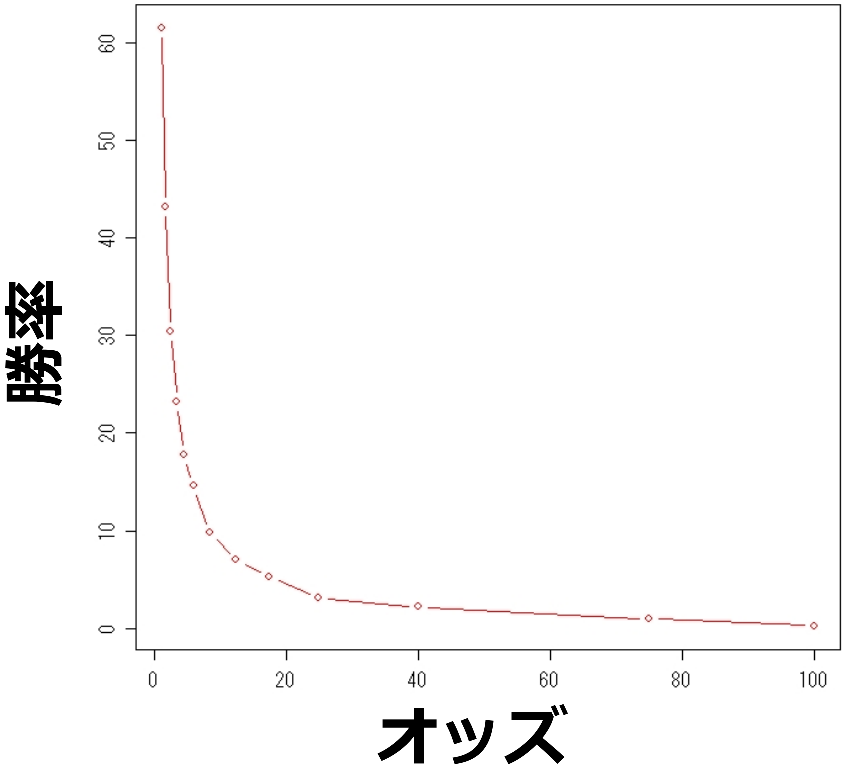
競馬のオッズ

競馬の回帰分析

回帰分析のオッズとは
ちょっと違う!!!

日本の競馬では、
賭けた金が何倍になって払い戻されるかをオッズと呼ぶ。





https://www.wikihouse.com/mamedalove/index.php?%BB%D9%BB%FD%CE%A8%A4%C8%BE%A1%CE%A8%A4%CE%B4%D8%B7%B8%A4%C8%A4%CF%3F

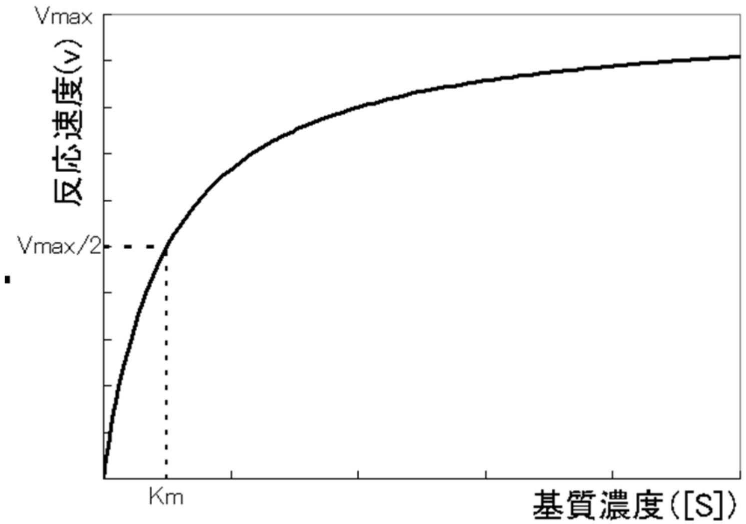
非線形回帰

ミカエリス・メンテンモデル

酵素の反応速度論に対するモデル

$$f(x,\beta)=\frac{\beta_1 x}{\beta_2+x}$$

$f(x,\beta)$: 酵素の反応速度
x ：基質濃度



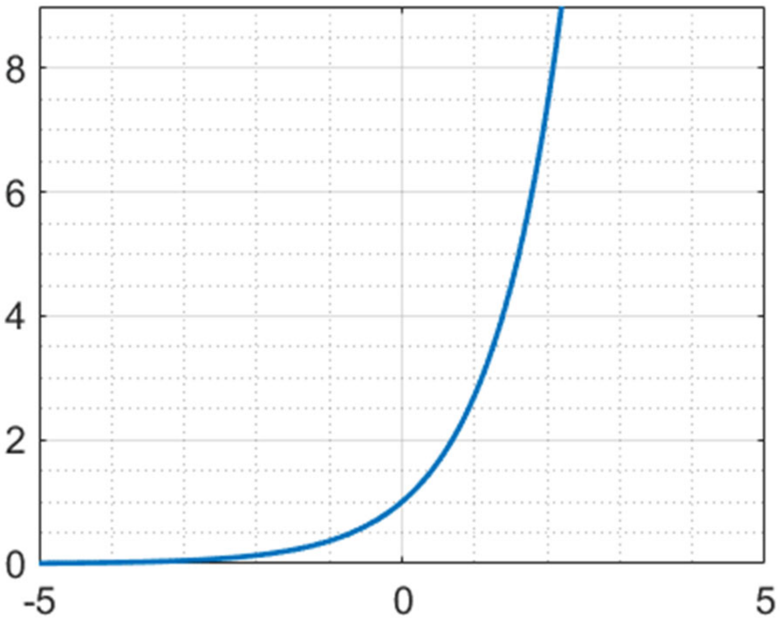
ミカエリス・メンテン式のv-[S]プロット

$$Y=\beta_o+\beta_1 X_1+\beta_2 X_2+\beta_3 X_3+\cdots.+\beta_p X_p+\epsilon$$

といったβ_iの線形結合で表せないため、非線形モデルになる

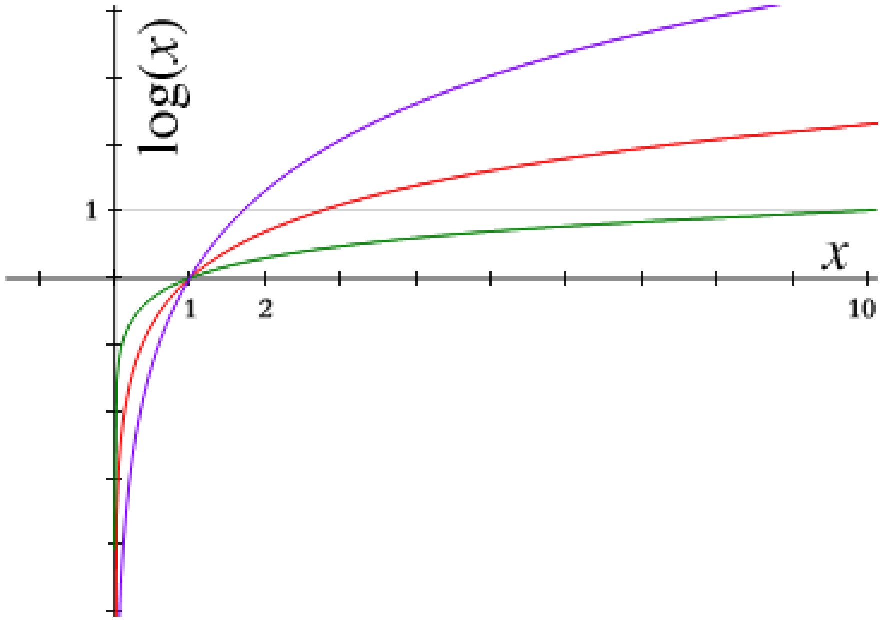
非線形回帰いろいろ

■ 指数関数　$y = e^{x}$



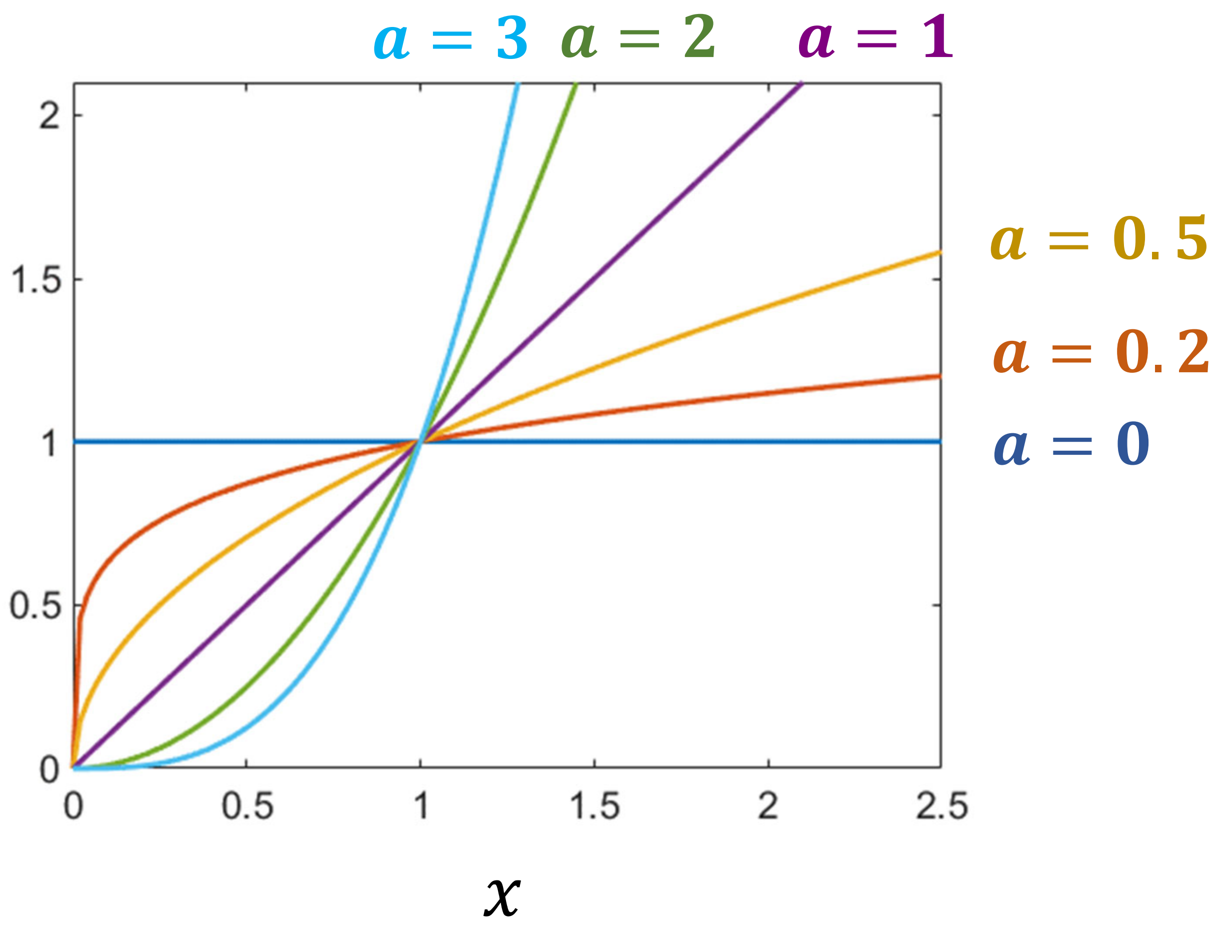
$$y = a^{x} \rightarrow \log_{\mathrm{a}} y = x$$

■ 対数関数　$y = a \log x$



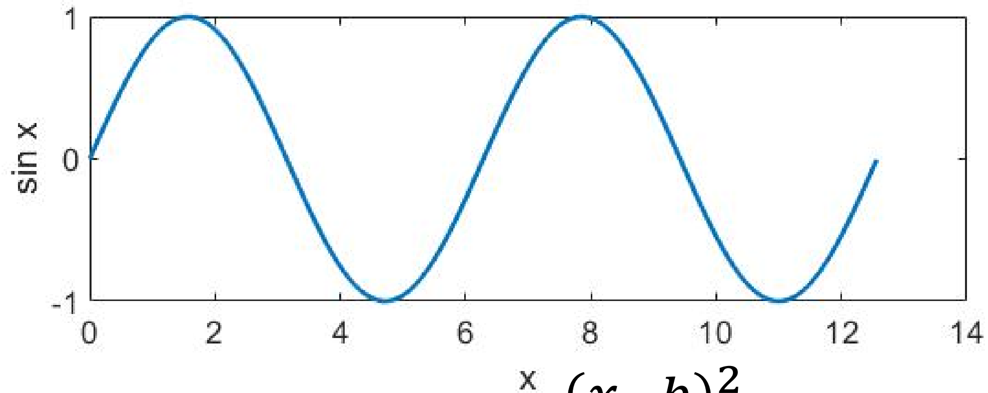
非線形回帰いろいろ

■ べき関数 $y = x^a = x * x * x * \cdots$

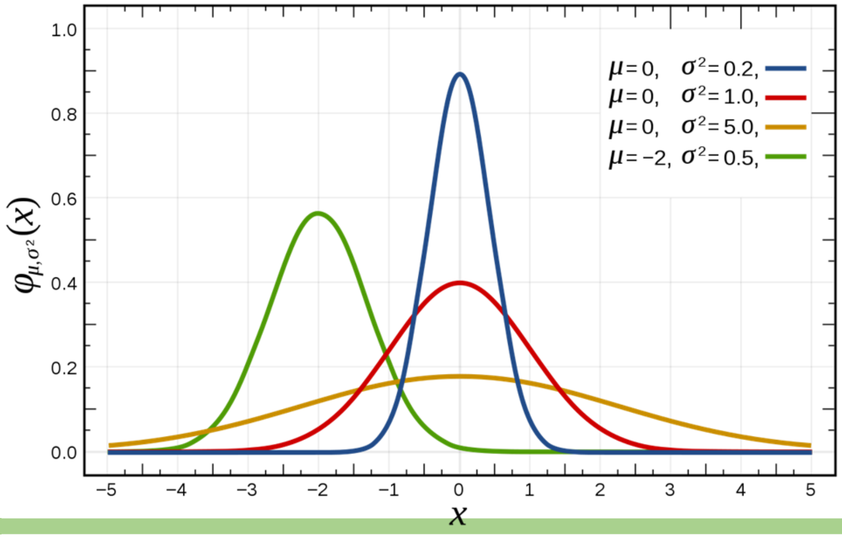


非線形回帰いろいろ

- 三角関数　$y = \sin x$



- ガウス関数　$y = a\exp(-\frac{(x-b)^2}{2c^2})$



復習

Scikit-learnで
回帰分析

いろいろな分析の共通点

復習

5週目：k-means法

6週目：単回帰分析

7週目：ロジスティック回帰分析

見た目全く違う解析ですが、
Scikit-learnを用いた解析という意味では、**手順がほとんど同じ**です。

ですので、k-means法で課題をきちんと解けなかった人は、
6週目、7週目の内容についていくのが難しくなります。

というわけで、手順の詳細をまとめてみました。

解析の手順

復習

1. scikit-learn のインポート
2. データの読み込み
3. headでデータの内容を確認
4. shapeでデータのサイズを確認
5. isnulで欠損値の確認→あれば削除(補完など。。)
6. インスタンスを作成
7. .fitを実行
8. .predictを実行

これはいつもの手順

大事なのはここ!!

(1) scikit-learn のインポート

復習

Numpy, Pandasの時と一緒

from sklearn.cluster import **KMeans**

from sklearn import **linear_model**

from sklearn.model_selection import **train_test_split**
from sklearn.linear_model import **LogisticRegression**

(2) データの読み込み

復習

乱数を使って
サンプルデータを
生成することもある。

■ CSV ファイルの読み込み

```
df = pd.read_csv('data/w5_rep_lattice.csv')
```

１列目をインデックスに入れるかどうか、オプションで指定できます。

https://note.nkmk.me/python-pandas-read-csv-tsv/

(2) データの読み込み

復習

■ Scikit-learnのサンプルデータの読み込み

アヤメのデータや、乳がんのデータなど、いろいろあります。

```
from sklearn import datasets
iris = datasets.load_iris()
```

このままだとirisはBunch型という変数になります。

```
iris_df = pd.DataFrame(iris.data, columns=iris.feature_names)
```

試しにtype(iris)　と実行してみましょう。
Bunch型のままでも解析できるのですが、
DataFrameで統一しました。

そのため、この行を使って、DataFrameに変換しています。

(3) headでデータの内容を確認

復習

df.head()

で、データの中身を確認。

どんな変数が格納されているか、見てみましょう。

何回もやりましたね!!!

(3) headでデータの内容を確認

復習

df.head()
で、データの中身を確認。
どんな変数が格納されているか、見てみましょう。

何回もやりましたね!!!

(4) shapeでデータのサイズを確認

それぞれの列にいくつNaNがあるか

df.shape
で確認。

(4) shapeでデータのサイズを確

復習

それぞれの列にいくつNaNがあるか

df.shape

で確認。

何回もやりましたね!!!

(5) 欠損値の確認→あれば削除

復習

それぞれの列にいくつNaNがあるか

df.isnull().sum()

欠損値が一つでもあれば、その行を削除

df = df.dropna(how='any')

本当は、この後、

df.isnull().sum()

をもう一回やって、欠損値が本当になくなったことを確認したほうがいい。

df.shape

をもう一回やって、欠損値削除後のデータサイズも確認したほうがいい。

復習

(５’) データの分割

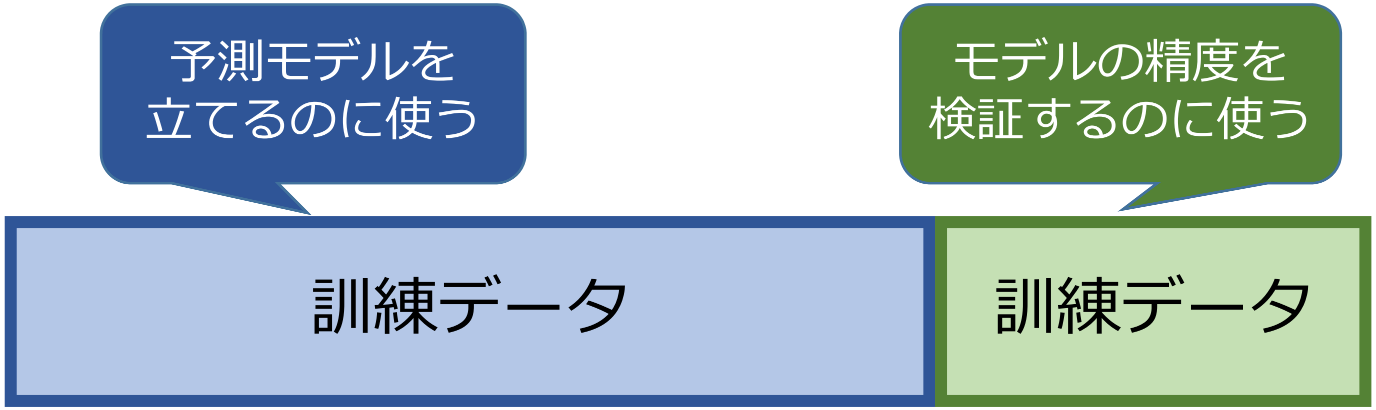
分割したデータを訓練データと検証データに分割

```
from sklearn.cross_validation import train_test_split

X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.3, random_state = 101)
```

X_train, Y_train: X座標、y座標の訓練データ
X_test, Y_test: X座標、y座標の検証データ

test_size = 0.3　は、全体の30%を検証用データとする、ということ。



(6)インスタンスを作成

復習

使う関数が違うだけ!!
オプションは関数によって、あったりなかったり。。

```
kmeans = KMeans(init='random',n_clusters=3)
```

```
model = linear_model.LinearRegression()
```

```
logmodel = LogisticRegression()
```

（7）.fitを実行

復習

#k-means法を実行

```
kmeans.fit(X)
```

単回帰分析を実行

```
model.fit(x, y)
```

xが1列だと単回帰分析
xが複数列あれば重回帰分析

ロジスティック回帰分析を実行

```
logmodel.fit(X_train,Y_train)
```

(8) .predictを実行

復習

クラスター番号を予測 / Predict the cluster number.

```
y_pred = kmeans.predict(X)
```

回帰直線を求める→説明変数から、目的変数を予測 / Predict the objective var.

```
reg_y = model.predict(x)
```

その事象が起こるか起こらないかを予測 / Predict the incidence

```
predictions = logmodel.predict(X_test)
```

(9) 解析後

復習

k-means法

分類したデータを可視化して確認
→クラスタごとに色分けしてプロット

単回帰分析

決定係数から、予測モデルの精度を確認

ロジスティック回帰分析

混合行列から、予測の精度を確認

ロジスティック回帰分析

$$y = \frac{1}{1 + \exp(-(\boldsymbol{\beta_0} + \boldsymbol{\beta_1} x_1 + \cdots + \boldsymbol{\beta_i} x_i))}$$

- $\beta_0, \beta_1, \ldots$を求める。
- どれくらい実データと相関があるか、検証する

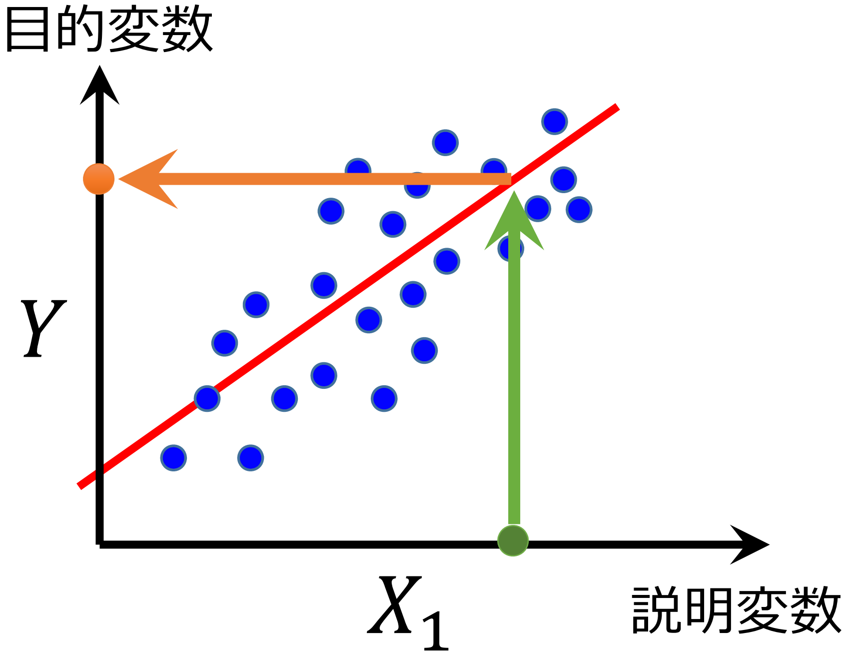
結果の見方

.coef_ ： 偏回帰係数$\beta_1,\ \beta_2,\ \beta_3,\ \ldots$

.intercept_ :切片β_0

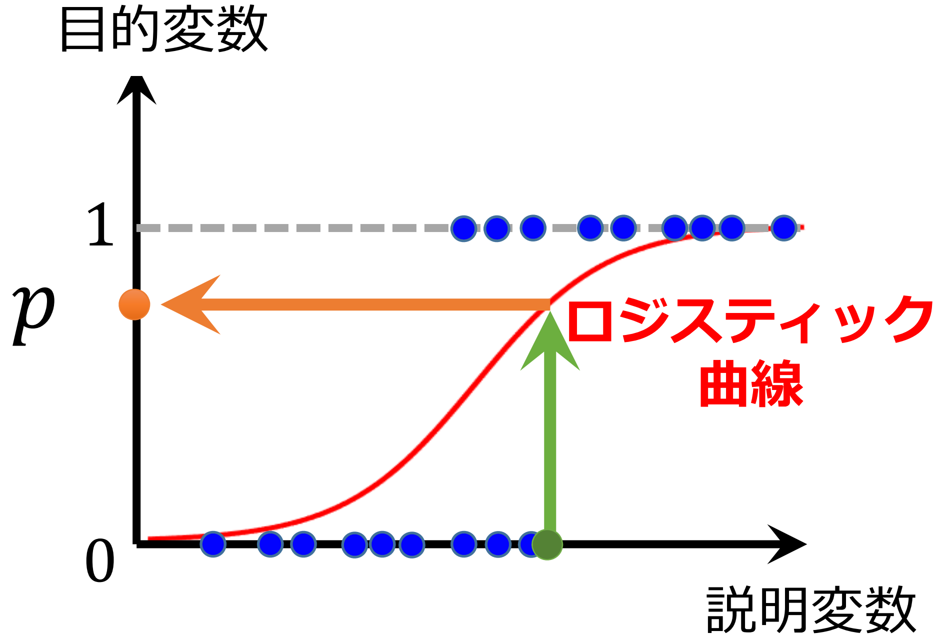
ロジスティック回帰分析

線形回帰分析
→**量的変数**の予測



$$Y = \boldsymbol{\beta}_o + \boldsymbol{\beta}_1 X_1$$

ロジスティック回帰分析
→**発生確率**の予測



$$p = \frac{1}{1 + \exp(-(\boldsymbol{\beta}_0 + \boldsymbol{\beta}_1 x_1))}$$

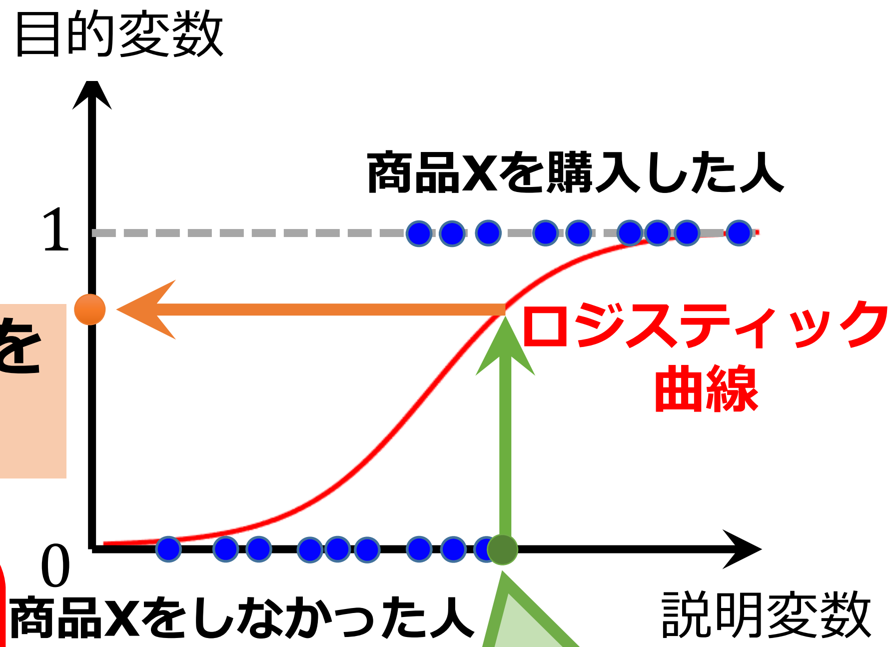
ロジスティック回帰分析

ある事象が **起きた→ 1**
起きなかった→0

ロジスティック回帰分析
→**発生確率**の予測

説明変数
- x_1 ：購入履歴
- x_2 ：その商品のサイトの閲覧回数
- x_3 ：年収
- x_4 ：職業
- x_5 ：家族構成

Aさんが商品Xを購入する確率



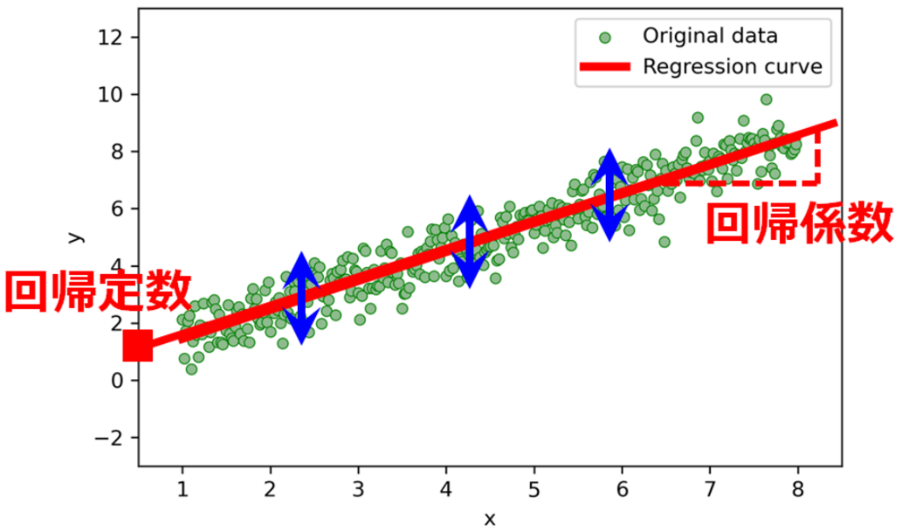
モデル化するとき
変数を標準化したら、
発生確率を計算するときは
標準化した値を使う!!

Aさんの入力データ
$\beta_0 + \beta_1 \boxed{x_1} + \beta_2 \boxed{x_2} + \cdots$

単回帰分析

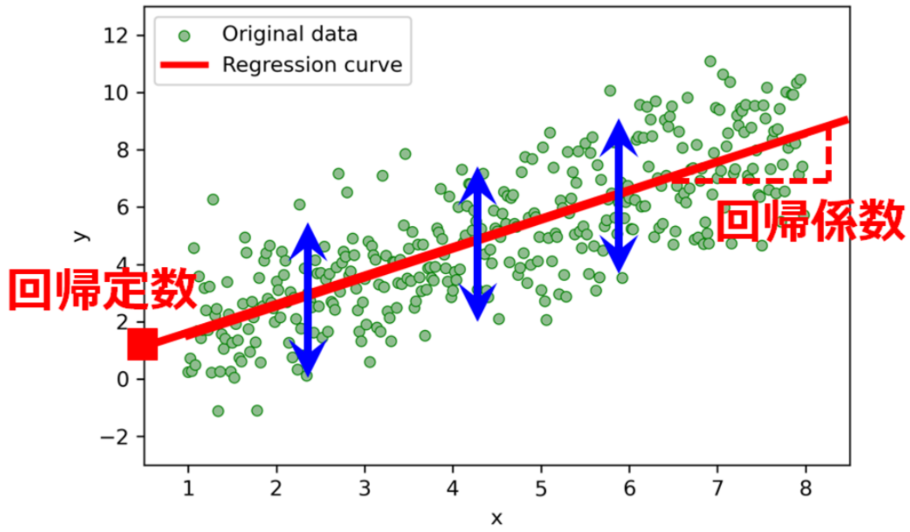
復習

$\sigma = 0.5$



決定係数0.92

$\sigma = 1.5$



決定係数0.63

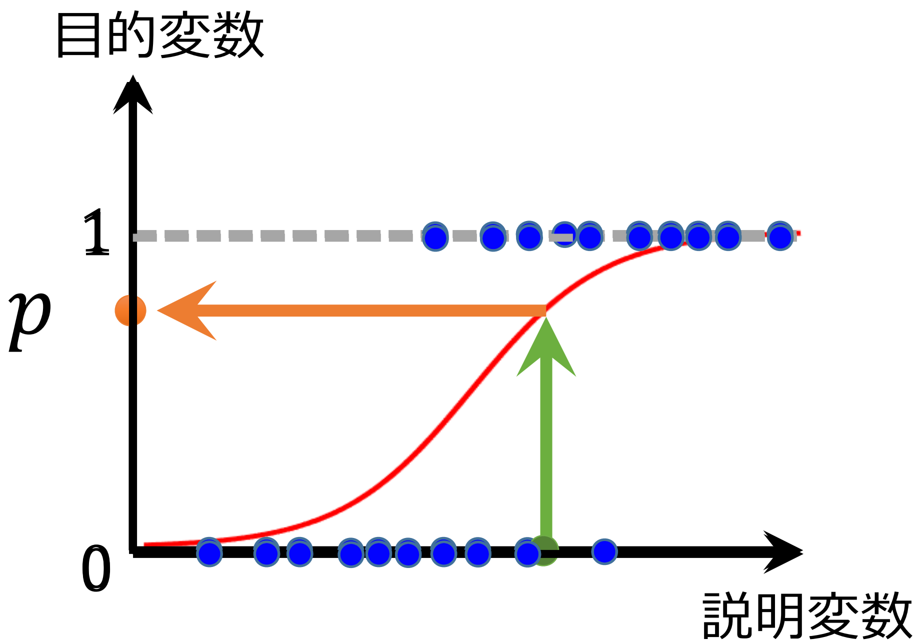
$$R^2 = 1 - \frac{\sum(y_i - f(x_i))^2}{\sum(y_i - \bar{y})^2}$$

ばらつきが小さいほうが、
右辺第2項が小さくなるので、
決定係数は大きくなる

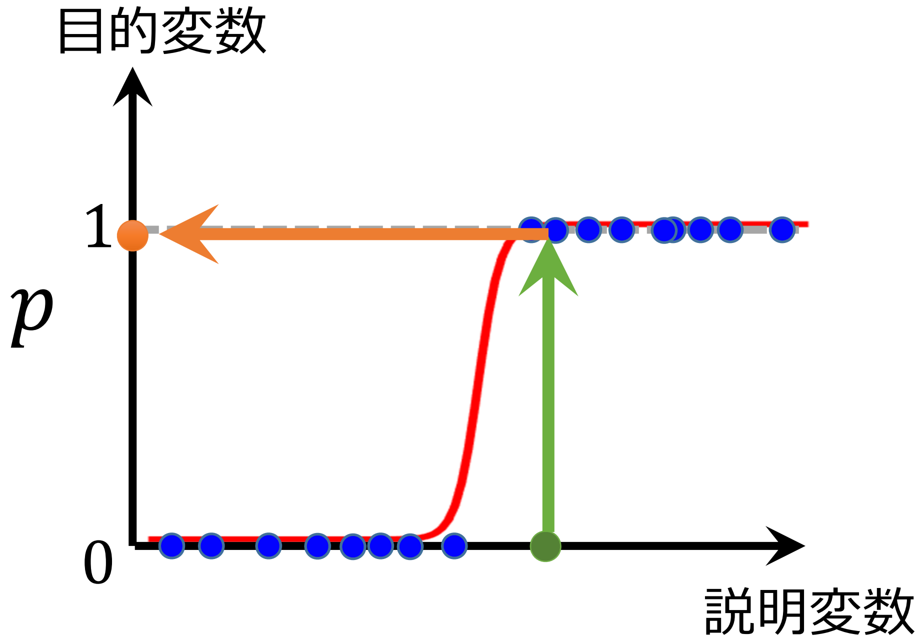
モデルの精度

$$p = \frac{1}{1 + \exp(-(\boldsymbol{\beta}_0 + \boldsymbol{\beta}_1 x_1))}$$

$\boldsymbol{\beta}_1$が小さい



$\boldsymbol{\beta}_1$が大きい



混合行列 (confusion matrix)

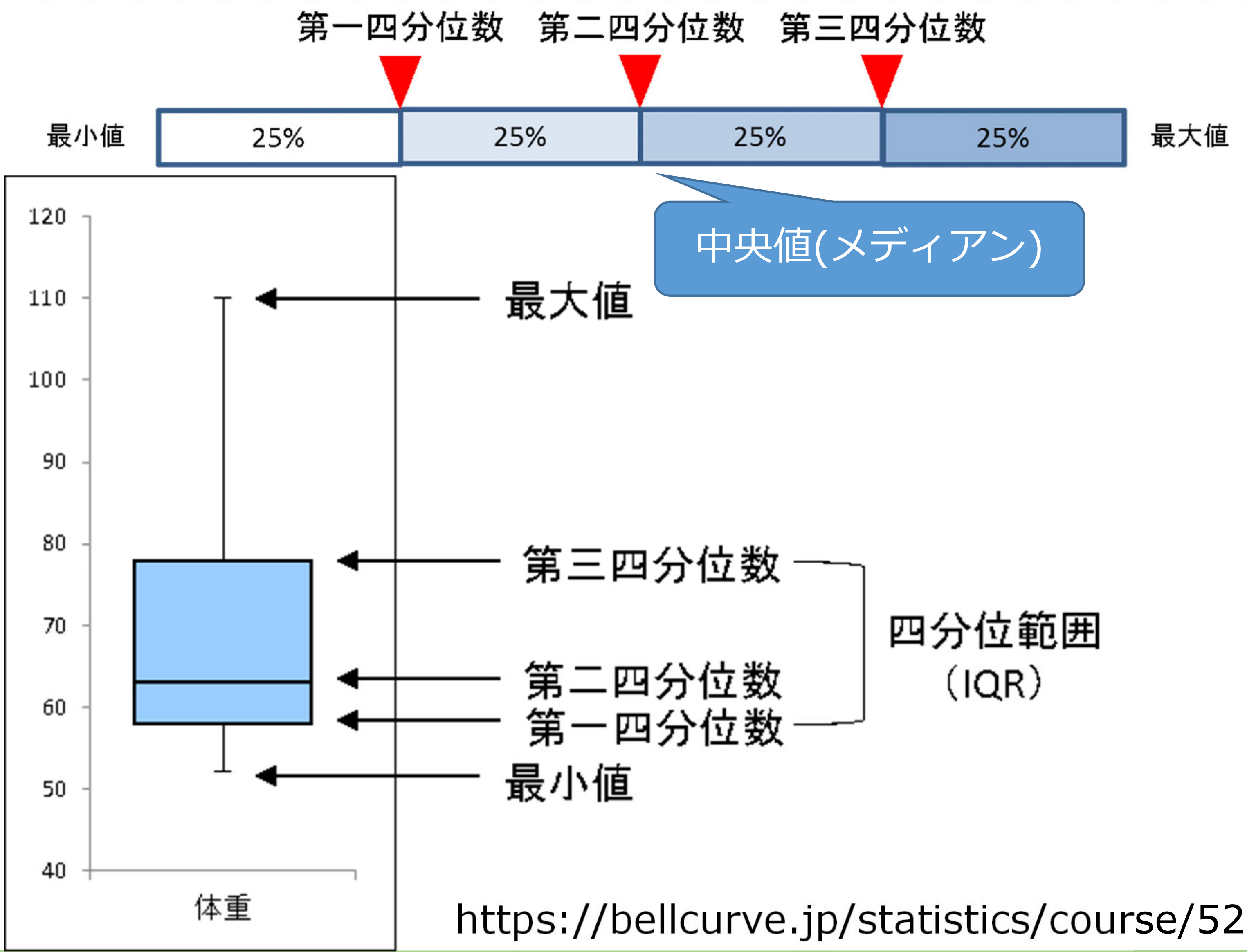
		予測	
		陽性	陰性
実際	陽性	True positive 真陽性	False positive 偽陽性
	陰性	False negative 偽陰性	True negative 真陰性

モデルの性能評価

正解率(Accuracy)$=\dfrac{FT+FN}{Total}$

検出率、**精度**など
他にもあります。

データ可視化：箱ひげ図



https://bellcurve.jp/statistics/course/5220.html

Pandas:get_dummies

sex = pd.**get_dummies**(dataset["sex"])

	female	male
1	1	0
3	1	0
6	0	1
10	1	0
11	1	0

文字列を数字に変換

sex = pd.**get_dummies**(dataset["sex"],**drop_first=True**)

	male
1	0
3	0
6	1
10	0
11	0

情報が重複するので
一列目を削除